AMENDMENTS TO THE CLAIMS

1.   (Currently amended) A computer network system having a fault management architecture ~~for use in a computer system,~~ as in Claim 61, the architecture comprising:

  a fault manager suitable for interfacing with diagnostic engines and fault correction agents, the fault manager being suitable for receiving error information and passing this information to the diagnostic engines that have subscribed to receive the error information;

  at least one diagnostic engine for receiving error information and identifying a set of fault possibilities associated with ~~the~~ errors contained in the error information;

  at least one fault correction agent for receiving the set of fault possibilities from the at least one diagnostic engine and then selecting a diagnosed fault, and then taking appropriate fault resolution action concerning the selected diagnosed fault; and logs for tracking ~~the~~ a status of the error information, ~~the~~ a status of the fault management exercises, and ~~the~~ a fault status of the resources of the computer system.

2.   (Original) The fault management architecture of Claim 1 wherein the fault manager is configured to accommodate additional diagnostic engines and fault correction agents that can be added at a later time.

3.   (Original) The fault management architecture of Claim 2 wherein the fault manager is configured so that said additional diagnostic engines and additional fault correction agents can be added while the computer system is operating without interrupting its operation.

4.   (Original) The fault management architecture of Claim 1 wherein the fault correction agents resolve faults by initiating at least one of: executing a

corrective action on a selected diagnosed fault and generating a message
identifying the selected diagnosed fault so that further action can be taken.

5.   (Currently amended) The fault management architecture of Claim 4 wherein
generating a message identifying the selected diagnosed fault so that further
action can be taken includes identifying faulted resource and identifying ~~the~~ a
problem with the faulted resource.

6.   (Original) The fault management architecture of Claim 1 wherein the
architecture further includes a data capture engine configured to obtain error
information from the computer system and generate an error report that is
provided to the fault manager.

7.   (Original) The fault management architecture of Claim 1 wherein the
diagnostic engine determines a probability of occurrence associated with each
identified fault possibility.

8.   (Original) The fault management architecture of Claim 7 wherein the at least
one fault correction agent for receiving the set of fault possibilities receives a
relative probability of occurrence associated with each identified fault possibility
from the diagnostic engines and then resolves a fault using a protocol.

9.   (Original) The fault management architecture of Claim 8 wherein the at least
one fault correction agent resolves a set of fault possibilities using a protocol that
incorporates at least one of: an analysis of at least one of computer resource
failure history, system management policy, and relative probability of
occurrence for each fault possibility.

10.   (Currently amended) The fault management architecture of Claim 1 wherein
the fault manager publishes ~~the~~ error reports; and wherein each diagnostic engine
subscribes to selected error reports associated with the fault diagnosis
capabilities of said diagnostic engine so that when the fault manager publishes

error reports <u>only</u> subscribing diagnostic engines receive the selected error
reports.

11.   (Original) The fault management architecture of Claim 1 wherein the fault
manager stores provided error reports in a log comprising an error report log and
wherein the error report log tracks the status of the provided error reports.

12.   (Original) The fault management architecture of Claim 6 wherein the
diagnostic engines and the agents are configured so that the fault manager
continuously accumulates error reports from the data capture engine until
enough error information is accumulated so that the diagnostic engines and the
agents can successfully diagnose a fault associated with the error reports.

13.   (Currently amended) The fault management architecture of Claim 6 wherein
the fault manager stores the error reports generated by the data capture engine to
~~the~~ an error report log of the logs;

wherein the at least one diagnostic engine stores fault management
exercise information in a fault management exercise log of the logs; and
wherein the at least one fault correction agent stores fault status
information concerning resources of the computer system in a resource cache of the
logs.

14.   (Currently amended) The fault management architecture of Claim 13 wherein
~~the~~ information from the error report log and the fault management exercise log
are stored in the resource cache.

15.   (Original) The fault management architecture of Claim 14 wherein resource
cache is configured so that in the event of a computer system failure, the system
can be restarted and information can be downloaded from the resource cache to

reconstruct error history, fault management exercise history, and resource status, and use this information to conduct fault diagnosis.

16.   (Original) The fault management architecture of Claim 14 wherein resource cache is configured so that in the event of a computer system failure, the system can be restarted and information can be uploaded from the resource cache, the error report log, and

the fault management exercise log to reconstruct error history, fault management exercise history, and resource status, and use this information to conduct fault diagnosis.

17.   (Currently amended) The fault management architecture of Claim 1 wherein the fault manager includes a soft error rate discriminator that:

receives error information concerning ~~correctible~~ correctable errors;

wherein the soft error rate discriminator is configured so that when the number and frequency of ~~correctible~~ correctable errors exceeds a predetermined threshold number of correctable errors over a predetermined threshold amount of time, these errors are deemed recurrent ~~correctible~~ correctable errors that are sent to the diagnostic engines for further analysis;

wherein the diagnostic engine receives a recurrent correctible error message and

diagnoses a set of fault possibilities associated with the recurrent correctible error message; and

wherein a fault correction agent receives the set of fault possibilities from the diagnostic engines and then resolves the diagnosed fault.

18.   (Currently amended) The fault management architecture of Claim 17 wherein the soft error rate discriminator receives error information concerning ~~correctible~~ <u>correctable</u> errors from the diagnostic engine.

19.   (Currently amended) The fault management architecture of Claim 17 wherein the diagnostic engine that identifies a set of fault possibilities associated with the recurrent ~~correctible~~ <u>correctable</u> error message further determines associated probabilities of occurrence for the set of fault possibilities associated with the recurrent ~~correctible~~ <u>correctable</u> error message.

20.   (Original) The fault management architecture of Claim 19 wherein the a fault correction agent receives the set of fault possibilities and associated probabilities of occurrence from the diagnostic engines and the agent then takes appropriate action to resolve the set of fault possibilities.

21.   (Currently amended) The fault management architecture of Claim 1 wherein the fault manager includes a soft error rate discriminator that:

        receives error information concerning soft errors;

        wherein the soft error rate discriminator is configured so that when the number and frequency of soft errors exceeds a predetermined threshold number of soft errors over a predetermined threshold amount of time, these soft errors are deemed recurrent soft errors that are sent to the diagnostic engines for further analysis;

        wherein the diagnostic engine receives a recurrent soft error message and diagnoses a set of fault possibilities associated with the recurrent ~~correctible~~ <u>correctable</u> error message; and

        wherein a fault correction agent receives the set of fault possibilities from the diagnostic engines and then resolves the diagnosed fault.

22.   (Original) The fault management architecture of Claim 1 further including a
fault management administrative tool that is configured to enable a user to
access the logs to determine the fault status and error history of resources in the
computer system.

23.   (Original) The fault management architecture of Claim 1 further including a
fault management statistical file that can be reviewed to determine the
effectiveness of the diagnostic engines and fault correction agents at diagnosing
faults and resolving faults.

24.   (Original) The fault management architecture of Claim 1 wherein the
computer system comprises a single computer device.

25.   (Original) The fault management architecture of Claim 1 wherein the
computer system comprises a plurality of computers forming a network.

26.   (Currently amended) A method for diagnosing and correcting faults in a
computer system having a fault management architecture; the method
comprising:

          receiving error information in a fault manager of the computer system;
          diagnosing a set of fault possibilities associated with the error
information, wherein said diagnosing is accomplished by the computer system; ~~and~~
          resolving the set of set of fault possibilities by choosing a selected fault
from among the set of fault possibilities and then resolving the selected fault,
wherein said choosing and resolving is accomplished by the computer system; and
          publishing error reports which are receivable by diagnostic engines that
have subscribed to receive the error information.

27.   (Original) A method as in Claim 26 wherein the receiving error information
in a fault manager of the computer system further includes:

capturing error information from the computer system;

generating an error report that includes the captured error information;

and providing the error report to the fault manager of the computer system.

28.   (Currently amended) A method as in Claim ~~26~~ 27 wherein capturing error information from the computer system includes capturing enough error information to enable a diagnosis of a fault to be made.

29.   (Original) A method as in Claim 26 wherein diagnosing a set of fault possibilities associated with the error information includes:

determining a set of fault possibilities associated with the error information and

determining a relative probability of occurrence for each fault possibility to generate a certainty estimation for each fault possibility.

30.   (Original) A method as in Claim 26 wherein choosing the selected fault associated with the error information is accomplished by implementing a computerized determination of a most likely fault associated with the error information.

31.   (Original) A method as in Claim 30 wherein choosing the selected fault by implementing a computer determination of a most likely fault associated with the error information includes an analysis of at least one of: computer resource failure history, system management policy, and relative probability of occurrence for each fault possibility.

32.   (Original) A method as in Claim 26 wherein resolving the diagnosed fault is accomplished by implementing computerized instructions that accomplish at least one of correction of the fault and generating a fault message that can be used to identify the fault and to take further action.

33.   (Original) A method as in Claim 26 wherein resolving the diagnosed fault is accomplished by implementing computerized instructions that accomplish at least one of software correction of the fault, software compensation for the fault, and generating a fault message that can be used to identify the fault and to take further action.

34.   (Original) A method as in Claim 26 wherein resolving the diagnosed fault is accomplished by implementing computerized instructions that accomplish at least one of software correction of the fault and software compensation for the fault.

35.   (Original) A method as in Claim 26 wherein the method further includes updating error logs to track each new error;

updating fault management exercise logs to track the current status of fault identification and fault diagnosis tracking error information; and

updating a resource cache to track the current fault status and fault history of resources of the computer system.

36.   (Original) A method as in Claim 35 wherein the resource cache includes elements of the error logs and the fault management exercise logs.

37.   (Original) A method as in Claim 26 wherein the method further includes: providing logs for at least one of tracking errors in the system, tracking the current status of fault diagnosis, tracking the current fault status of a resource of the computer system; and tracking a fault history of a resource of the computer system; and updating the logs based on changes in status.

38.   (Original) A method as in Claim 37 wherein, if the computer system shuts down due to an error, the method comprises the further steps of:

restarting the system;

recalling the logs to track the fault status and fault history of resources of the computer system and thereby diagnose a fault; and

resolving the fault.

39.   (Currently amended) A computer-readable program product for diagnosing and correcting faults in a computer system having a fault management architecture as in Claim 61, the computer-readable program product configured to cause a computer to implement the computer-controlled steps of:

receiving error information in a fault manager of the computer system;

diagnosing a set of fault possibilities associated with the error information;

choosing a selected fault possibility from among the set of fault possibilities; and

resolving the selected fault possibility to resolve a fault; and

publishing error reports which are receivable by diagnostic engines that have subscribed to receive the error information.

40.   (Original) A computer-readable program product as in Claim 39 wherein the computer controlled step of receiving error information in a fault manager of the computer system further includes computer readable instructions for:

capturing error information from the computer system;

generating an error report that includes the captured error information; and providing the error report to the fault manager of the computer system.

41.   (Original) A computer-readable program product as in Claim 40 wherein the computer system incorporates diagnostic engines to diagnose faults based on error information and wherein the computer-controlled step of capturing error information includes capturing enough error information to enable a diagnosis engine to diagnose a fault.

42.   (Original) A computer-readable program product as in Claim 39 wherein the computer-controlled step of diagnosing a set of fault possibilities associated with the error information includes:

   determining a set of fault possibilities associated with the error information and determining a relative probability of occurrence for each fault possibility.

43.   (Original) A computer-readable program product as in Claim 39 wherein the computer-controlled step of choosing a selected fault from among the set of fault possibilities is accomplished by implementing computer readable instructions for determining a most likely fault possibility associated with error information.

44.   (Original) A computer-readable program product as in Claim 43 wherein determining the most likely fault associated with error information includes an analysis of at least one of: computer resource failure history, system management policy, and relative probability of occurrence for each fault possibility.

45.   (Original) A computer-readable program product as in Claim 39 wherein the computer-controlled step of resolving the diagnosed fault is accomplished by implementing computer readable instructions for accomplishing at least one of: correcting the fault and generating a fault message that can be used to identify the fault and be used to take further action.

46.   (Original) A computer-readable program product as in Claim 39 wherein the product further includes computer readable instructions for generating logs that enable at least one of: tracking error information received by the system;

   tracking the current status of fault diagnosis; tracking the current fault status of a resource of the computer system; and tracking a fault history of a resource of the computer system; and

updating the logs based on changes in status.

47.   (Original) A computer-readable program product as in Claim 46 wherein the
product further includes computer readable instructions that, if the computer
system shuts down due to an error, further comprise computer readable
instructions for:

restarting the system;

recalling the logs to reestablish the fault status and fault history of
resources of the computer system and thereby diagnose a fault; and resolving the
fault.

48.   (Currently amended) A computer system comprising:

a processor capable of processing computer readable instructions and
generating error information;

a memory capable of storing computer readable information;

computer readable instructions enabling the computer system to capture
error information from the computer system and generating error reports;

computer readable instructions enabling the computer system to analyze
the error reports and generate a list of fault possibilities associated with the error
reports;

computer readable instructions enabling the computer system to
determine a probability of occurrence associated with each of the fault possibilities;

computer readable instructions enabling the computer system to
determine which of the of fault possibilities is the most likely to have caused the
error report and select that as an actionable fault;

computer readable instructions enabling the computer system to resolve
the actionable fault; ~~and~~

computer readable instructions enabling the computer system to
understand that the actionable fault has been resolved<u>; and</u>

computer readable instructions enabling the computer to pass the error

reports to diagnostic engines that have subscribed to receive the error reports.

49.   (Currently amended) The computer system of <u>claim</u> 48 further including

computer readable instructions enabling the computer system to generate an

error log that includes a listing of error reports.

50.   (Currently amended) The computer system of <u>claim</u> 48 further including

computer readable instructions enabling the computer system to generate a fault

management exercise log that includes a listing of fault possibilities and the

current status of fault diagnosis.

51.   (Currently amended) The computer system of <u>claim</u> 48 further including

computer readable instructions enabling the computer system to generate an

automatic system recovery unit log that includes a listing of the current fault

status of system resources of the computer system, a listing of fault diagnosis

concerning the system resources, and a listing of error reports that led to the of

fault diagnosis concerning the system resource;

wherein, in the event of computer system failure, upon system restart, the

information in the automatic system recovery unit log can be recalled and analyzed

to diagnose faults.

52.   (Currently amended) A computer network system having a fault management

architecture configured for use in a computer system, the computer network

system comprising:

a plurality of nodes interconnected in a network;

a fault manager mounted at a first node on the network and configured to

diagnose and resolve faults occurring at said first node, the fault manager being

suitable for receiving error information and passing this information to diagnostic engines that have subscribed to receive the error information.

53.   (Currently amended) A computer network system having a fault management architecture as in Claim 52, wherein the fault manager is configured to interface with diagnostic engines and fault correction agents, and is suitable for receiving error information and passing this information to the diagnostic engines;

the fault manager including:

at least one diagnostic engine for receiving error information from the first node and

diagnosing a set of fault possibilities associated with ~~the~~ errors contained in the error information;

at least one fault correction agent for receiving the set of fault possibilities from the at least one diagnostic engine and then selecting a diagnosed fault from among the set of fault possibilities, and taking appropriate fault resolution action concerning the selected diagnosed fault; and

logs for tracking ~~the~~ a status of <u>the</u> error information, ~~the~~ a status of <u>the</u> fault management exercises, and ~~the~~ a fault status of <u>the</u> resources of the first node.

54.   (Currently amended) <u>A computer network having a</u> ~~The~~ fault management architecture of Claim 53 wherein the fault manager is configured so that said additional diagnostic engines and additional fault correction agents can be added to the fault manager while the computer system is operating without interrupting the operation of the network.

55.   (Currently amended) <u>A computer network having a</u> ~~The~~ fault management architecture of Claim 53 wherein the fault manager includes a soft error rate discriminator that:

receives error information concerning soft errors;

wherein the soft error rate discriminator is configured so that when the number and frequency of soft errors exceeds a predetermined threshold number of soft errors over a predetermined threshold amount of time, these errors are deemed recurrent soft errors that are sent to the diagnostic engines for further analysis;

wherein the diagnostic engine receives a recurrent soft error message and diagnoses a set of fault possibilities associated with the recurrent soft error message; and wherein a fault correction agent receives the set of fault possibilities from the diagnostic engines and then resolves the diagnosed fault.

56.   (Original) A computer network system having a fault management architecture as in Claim 52, wherein the fault manager mounted at a first node on the network is configured to diagnose and resolve faults occurring at other nodes of the network.

57.   (Currently amended) A computer network system having a fault management architecture as in Claim 56, wherein the fault manager is configured to interface with diagnostic engines and fault correction agents, and is suitable for receiving error information and passing this information to the diagnostic engines;

the fault manager including:

at least one diagnostic engine for receiving error information from the nodes of the network and diagnosing a set of fault possibilities associated with the errors contained in the error information;

at least one fault correction agent for receiving the set of fault possibilities from the at least one diagnostic engine and then selecting a diagnosed fault from among the set of fault possibilities, and taking appropriate fault resolution action concerning the selected diagnosed fault; and

logs for tracking ~~the~~ a status of error information, ~~the~~ a status of ~~the~~ fault management exercises, and ~~the~~ a fault status of ~~the~~ resources of the nodes of the network.

58.   (Currently amended) <u>A computer network having a</u> ~~The~~ fault management
architecture of Claim 56 wherein the fault manager is configured so that said
additional diagnostic engines and additional fault correction agents can be added
to the fault manager while the computer system is operating without interrupting
the operation of the network.

59.   (Currently amended) <u>A computer network having a</u> ~~The~~ fault management
architecture of Claim 56 wherein the fault manager includes a soft error rate
discriminator that:

   receives error information concerning soft errors;

   wherein the soft error rate discriminator is configured so that when the
number and frequency of soft errors exceeds a predetermined threshold number of
soft errors over a predetermined threshold amount of time, these errors are deemed
recurrent soft errors that are sent to the diagnostic engines for further analysis;
wherein the diagnostic engine receives a recurrent soft error message and diagnoses a
set of fault possibilities associated with the recurrent soft error message; and wherein
a fault correction agent receives the set of fault possibilities from the diagnostic
engines and then resolves the diagnosed fault.

60.   (New) A fault management architecture for use in a computer system, the
architecture comprising:

   a fault manager suitable for interfacing with diagnostic engines and fault
correction agents, the fault manager being suitable for receiving error information
and passing this information to the diagnostic engines;

   at least one diagnostic engine for receiving error information and
identifying a set of fault possibilities associated with errors contained in the error
information;

at least one fault correction agent for receiving the set of fault possibilities from the at least one diagnostic engine and then selecting a diagnosed fault, and then taking appropriate fault resolution action concerning the selected diagnosed fault;

logs for tracking a status of the error information, a status of the fault management exercises, and a fault status of the resources of the computer system; and

wherein the fault manager publishes error reports; and wherein each diagnostic engine subscribes to selected error reports associated with the fault diagnosis capabilities of said diagnostic engine so that when the fault manager publishes error reports only subscribing diagnostic engines receive the selected error reports.

61.   (New) A computer network system having a fault management architecture configured for use in a computer network system, the computer network system comprising:

a plurality of nodes interconnected in a network; and

a fault manager mounted at a node on the network and configured to diagnose and resolve faults occurring at said node.